Macromedia Flash MX 2004 and Video

by Forest Key and Chris Hock

February 2004

by Forest Key and Chris Hock

Contents

Executive Summary

This white paper provides a general overview of the video capabilities of the Macromedia Flash MX 2004 platform.

Introduction

Video and the Internet seem like a match made in heaven. Video is the rich media medium that most closely simulates our day-to-day visual experiences. The Internet is a boundless playground to search for and consume interesting content. Thousands of compelling websites should integrate video with data, content and interactive controls to create rich experiences that go way beyond what is possible with a static television set… right?

Unfortunately early video content on the web has tended to be very static and television like—a rectangle of content playing back on your computer monitor, usually in a separate pop-up window covering the website pages that spawned it. The images were small, ugly, and the overall experience was poor.

Several technical challenges have kept designers from fully leveraging video content, including:

- **Bandwidth limitations.** Video is a data-intensive format, requiring megabytes of data to display even short video clips of less than one minute in length. The growth of broadband has greatly alleviated this technical obstacle, and increasingly large numbers of users have the bandwidth required to receive video content via the web.

- **Complexity of authoring video for the web.** There have been no standard tool sets to allow for creation of interactivity, navigation control, and fusion of video with other rich media content. Furthermore, most video playback clients are not pre-installed on most end users' systems, requiring lengthy downloads and a break in the overall immersive experience when visiting a website.

- **Lack of compelling integration of video and other web content.** Most video formats for the web offer no rich media capabilities beyond playback and display of video in a rectangular window. Video on the web needs to go beyond the television concept!

Macromedia Flash and Flash Video

Flash offers technological and creative benefits that free designers to create immersive rich experiences that fuse video together with data, graphics, sound, and dynamic interactive control. The advantages of using Flash include:

- **Immersive experiences.** Flash Video is just another media type within the Flash movie, allowing video to be layered, scripted, and controlled just like any other object in the Flash movie. Flash Video is an integral part of the experience, as opposed to a separate pop-up window that interrupts the viewing experience!

- **Custom branding and programmability.** Flash Video can easily be reskinned to convey custom branding and unique controls, and can dynamically adapt based on data-driven content (playlist, closed captioning, navigation, meta data, and so on).

- **Ubiquity.** Flash Video was introduced with Macromedia Flash Player 6 in the spring of 2002. Since then, Flash Player has become the most widely installed Internet video client, running over 90 percent of all Internet-connected PCs (source: NPD research April 2003). By comparison, Windows Media Player and Real Player have less than 60 percent market penetration, and QuickTime Player has less than 40 percent. This ubiquity ensures that Flash websites that use video will load quickly, with no need for additional plug-in downloads.

Technical Overview of Video Standards

Many people creating Flash Video projects files have a background in web design and desktop publishing, and might not have the technical understanding of video necessary to achieve optimal image quality for their video projects. This section explains the basic concepts and usage of video parameters and standards. If you are already familiar with these standards and the terminology, you can skip this section and continue with the section "Introducing the Macromedia Flash MX 2004 Video Platform."

Video Standards—NTSC and PAL

The video that you see on your television screen follows standards established in the 1950s when color television was first introduced. The leading formats in use today are NTSC (National Television System Committee) and PAL (Phase Alternating Line). Generally speaking, NTSC is the standard used in the Americas and Japan, whereas PAL is used in Europe, Australia, the Middle East, and Asia.

Neither video standard is optimal for presentation on computer monitors; each poses different challenges when you are trying to optimize video for web delivery:

- **Frame Size:** NTSC and PAL have different image sizes, which differ from the available image sizes of computer monitors.

- **Frame Rate:** NTSC and PAL have different frame rates for the display of images, which are different than those used by computer monitors.

- **Pixel Aspect Ratio:** NTSC and PAL share a pixel aspect ratio (referred to as *D1 Aspect Ratio*, which is essentially rectangular), but this ratio differs from that used by computer monitors (which is square).

- **Display:** NTSC and PAL consist of two separate "interlaced" fields, while computer monitors display "progressive" images.

2

Table 1 provides an overview of the differences between these standards.

Table 1: *Video Standards at a Glance*

	Image Size	Frame Rate	Aspect Ratio	Display
NTSC	720 x 480	29.97	D1	Interlaced
PAL	720 x 576	25	D1	Interlaced
Computer	Varies (much larger)	--	Square	Progressive

Frame Size

Conventional television screens are made up of horizontal lines while computer monitors consist of a series of horizontal and vertical pixels. The standard line resolution for an NTSC television is 525 lines; for PAL it is 576 lines. Most modern computer monitors have much higher vertical resolutions (measured in pixels), such as 768 or 1024, requiring vertical upscaling during playback in order to fill the monitor.

For NTSC video images, the SMPTE 259M professional standard specifies that the 525 lines be represented as 720 x 486—that is, 720 horizontal pixels by 486 vertical pixels. This default video size is commonly known as D1. Capturing footage with most modern video capture cards from a professional BetaSP or Digital Betacam source result in a D1-sized frame. Capturing footage from a DV source, however, yields a 720 x 480 frame. The difference between the D1 spec and the DV spec is only 6 vertical pixels. Many compression algorithms, including DV compression, prefer image sizes to be a multiple of 16. By shaving off the 6 pixels from a D1 resolution, the DV format was able to have a native resolution with a multiple of 16.

For PAL video images, frames are always 720 x 576, regardless of video source. Because PAL's vertical resolution, 576, is a multiple of 16, no change is necessary for DV compression.

Frame Rate

Video is essentially a sequence of images flashed on the screen in rapid succession, giving the illusion of motion. The number of frames displayed every second is known as the *frame rate*, and it is measured in *frames per second (fps)*. The higher the frame rate, the more frames per second will be used to display the sequence of images, resulting in smoother motion. The trade-off, however, is that higher frame rates require a higher amount of data—or system bandwidth—to display the video.

In a broad sense, NTSC video runs at 30 fps, and PAL runs at 25 fps. In actuality NTSC runs at 29.97 fps. The reason for the odd frame rate dates back to the transition from black and white television to color TV signals, where the 29.97 fps rate was chosen to ensure backwards compatibility with existing television sets. The fractional rate is more of a mathematical issue than anything else—there are still 30 frames, but they run 0.1 percent slower than actual time, giving you a frame rate of 29.97 fps.

When working with compressed movies in a format like Flash Video, the more frames that have to be displayed in a second the higher the file size. To manage the final file size, you have to lower either the frame rate or data rate. If you lower the data rate and leave the frame rate unchanged, the image quality is reduced to yield a smaller file size. If you lower the frame rate and leave the data rate unchanged, the file size is reduced but the movie might appear to stutter and motion may look less fluid than desired.

Whenever the frame rate is reduced, it is always a good idea to use an evenly divisible ratio of the original frame rate. If your source has a frame rate of 24 fps, then you should drop the frame rate to 12 fps, 8 fps, 6 fps, 4 fps, 3 fps, or 2 fps. If the source frame rate is 30 fps, in most cases you can adjust the frame rate to 30 fps, 15 fps, 10 fps, 6 fps, and so on. If your video is more than 10 minutes long, then audio will drift noticeably out of synch if you do not adhere to the 29.97 fps rate or an accurate even division for lower frame rates (such as 29.97/2 = 14.98).

Pixel Aspect Ratio

The D1/DV NTSC and PAL specification specify non-square pixels (often called *D1 aspect ratio*), while computer monitor pixels are square. D1 pixels are vertically shorter. For this reason when looking at a D1 video image on a computer monitor, the images will appear to be squashed vertically— making actors appear to be shorter. When this image is output and displayed on a broadcast monitor, the pixels are wider than they are tall and will show perfectly normal (see Figure 1).

Video Image on Video Monitor   Video Image on Computer Monitor

Figure 1: *The same video image displayed on a television monitor (left) and computer monitor (right). Note the image looks vertically compressed on the computer monitor, but normal on the television monitor.*

For this reason video images that are intended for display on computer monitors must be *pixel aspect corrected* by scaling the image to a valid 4:3 aspect ratio. For NTSC, the full square pixel resolution is 720 x 540 (vertical compensation), and for PAL it is 768 x 572 (horizontal compensation). Commonly used final video display resolutions on the Internet include 640 x 480, 512 x 384, 320 x 240, and 160 x 120.

Most video editing applications compensate for the pixel aspect ratio discrepancy by scaling the video image in real time while rendering it on the computer monitor. This is done because eventually the images are intended to return to television monitors for final display, and scaling the actual pixels in the video file would needlessly introduce a subtle distortion from the scaling operation. However, for web display, this real-time compensation is not a valid approach, given that the video sequence is destined to be

displayed on a square pixel monitor, and as such should be *hard-rendered* to compensate for the discrepancy.

Interlaced and Progressive Video

Video images consist of two *interlaced* fields that together comprise a frame (see Figure 2). This approach was introduced when TV was first invented due to a technical limitation that prevented a full frame to be "progressively" drawn on the monitor (from top to bottom) without a noticeable visual shuttering (as the images where being displayed it appeared as though they were being wiped on the screen). By breaking up the image into two fields (halves) and displaying one after the other this artifact was eliminated. This legacy technique has been a tremendous obstacle in the digital age of video and computers, and has been eliminated from newer video standards for High Definition television, which are *progressive* (images are drawn in one pass from top to bottom). Both interlaced groups of lines are known as a field, and are referred to as the *upper* field and the *lower* field. Fields are also sometimes referred to as Field 1 and Field 2, or odd and even, or top and bottom. Unfortunately there is not a standard nomenclature.

Figure 2: *Illustrating the effects of interlaced images.*

With real video footage, two interlaced fields often look very similar and no visible artifacts appear when looking at a video frame on a computer monitor. However, with video footage that includes high motion material that changes quickly (such as movement of the camera or of people in the frame) very noticeable field artifacts will appear giving the image a ghosted quality. This is due to the composition of two moments in time together in one frame.

In order to display crisp video on a computer monitor video frames must be *de-interlaced* by eliminating one of the fields. Half the information of each frame is discarded and the remaining information doubled or interpolated, in NTSC's case giving you 30 frames of 30 distinct points in time (see Figure 3).

Figure 3: *In this example the image on the left is an interlaced frame; in the image on the right one of the fields has been eliminated to produce a "de-interlaced" frame.*

Modern video standards for digital television have eschewed interlacing in favor of progressive scan display techniques. Progressive scan video cameras usually have the ability to switch back from progressive scan to interlaced video, and most of these cameras have a variety of frame rates with and without interlacing. Typical frame rates are described as 60p (60 fps progressive), 30i (30 fps interlaced), 30p (30 fps progressive), and 24p (24 fps progressive). When working with progressive images there is no need to de-interlace footage prior to deploying to the web.

Introducing the Macromedia Flash MX 2004 Video Platform

With the introduction of the Flash MX 2004 platform many new video capabilities and services have been added to the Flash video platform. Below is a list of Macromedia products used to create and deliver compelling Flash video experiences:

- **Flash Player 7** introduces greatly improved video quality through higher frame rates and improved image quality. Flash Video (FLV) files can now be dynamically loaded at runtime, permitting multimedia producers to use larger and longer video files within the Flash MX authoring environment. Note: Flash Player 6 supports streaming FLV files from Flash Communication Server.
- **Flash MX 2004** introduces the Video Import Wizard, which adds new options for encoding video on import into the Flash Timeline.
- **Flash MX Professional 2004** introduces Media Components, a set of components that enables users to incorporate external FLV files and to connect to Macromedia Flash Communication Server video streams; Flash Video Exporter, a new plug-in for use with third-party applications that enables users to encode audio and video into the FLV file format; and a series of behaviors that, together with slides, simplify and accelerate the creation of advanced interactive video presentations.
- **Flash Communication Server** [*available separately*] is a Macromedia streaming media server that streams audio and video to Flash Player 6 or higher.
- **Flash Video Streaming Service** [*available separately*] is a new service, based on Flash Communication Server, for quickly and easily streaming Flash video from a reliable, Content Delivery Network (CDN).

Approaches to Incorporating Video

The Flash MX 2004 products enable developers to use one of three techniques when delivering video:

- Embedding video within Flash Player movies
- Producing progressive download FLV files

6

- Producing streaming FLV files

Embedding Video within Flash Player Movies

Since the introduction of Flash MX and Flash Player 6, multimedia developers have been able to embed video within Flash Player movies by importing video and placing it on the Flash Timeline. A key benefit of this approach is the ability to see the individual video frames on the Flash Timeline and create overlays and interactions with the aid of the Flash design tools.

Flash MX 2004 builds upon this approach by introducing the Video Import Wizard, which provides fine control over encoding options, scaling and cropping presets, as well as color and brightness settings.

However, this approach is not without limitations:

- During authoring, each time you want to preview or test part or all of your Flash movie, you must publish the entire video file. This can add significant time to the authoring process.
- For web delivery, the entire video file must be downloaded from the web server in order for playback to begin.
- At runtime, the entire video file must fit into the local memory of the playback system.
- After approximately 120 seconds of continuous video playback, users may experience audio synch problems.
- File length is limited to a maximum duration of no greater than 16,000 frames.
- The video frame rate and Flash Timeline frame rate must be the same (because they share the same time base).

Producing Progressive FLV Files

Flash Player 7 introduces a new technique called *progressive download*, which enables developers to use ActionScript commands to feed external FLV files into a Flash movie and play them back during runtime. More specifically, you can use the netConnection and netStream commands to set the FLV file to play back, and to control the Play, Pause, Seek (to a timecode), Close behaviors and buffertime and size for a given video file.

Flash MX Professional 2004 also includes a set of components called Media Components that you can use to quickly add a full-featured FLV or MP3 playback control to your Flash project. Media Components provides support for both progressive download and streaming FLV files (see Figure 4). Flash MX Professional 2004 also includes a set of behaviors that can be used in conjunction with Media Components to create automated interactions between video sequences and slides in a project. (For details on using ActionScript and Media Components, see the reference guide, which you can access through the Flash MX 2004 Help panel.)
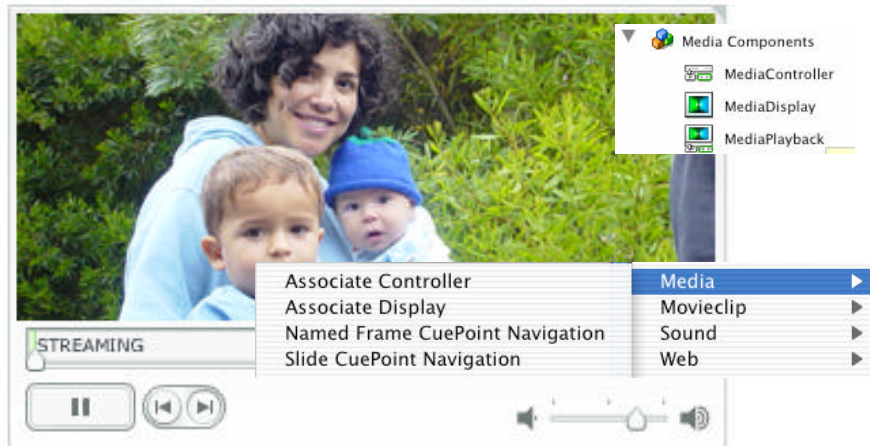
Figure 4: *Flash MX Professional 2004 Media Components support both progressive and streaming FLV files; to control display of slides you can use media behaviors.*

Using progressive FLV files has these advantages:

- During authoring, you need to publish only the Flash movie interface for previewing or testing part or all of your Flash movie. This results in faster preview times and quicker turnaround on iterative experimentation.

- For delivery, video begins playing as soon as the first segment has been downloaded and cached to local disk.

- At runtime, video files are streamed from the local disk into the SWF file, with no limitation on file size or duration. There are no audio synch issues or memory restrictions.

- The frame rate of the video file can be different from the frame rate of the SWF file, allowing for greater flexibility in setting up a project.

Producing Streaming FLV Files

Last, but hardly least, you can also stream video and audio files from Flash Communication Server (available separately). This approach provides efficient delivery of FLV (and MP3) files and offers these advantages:

- Less client resources (memory and disk space), since the clients do not need to download the entire file.

- More efficient use of network resources, since the only bits that are sent to the client are bits that are viewed.

- More secure delivery of media, because media does not get saved to the client's cache when streamed.

- Better tracking, reporting, and logging ability—key for industries such as video ad serving.

- Ability to deliver live video and audio, or to capture video from client's webcam or DV cameras.

- Multiway and multiuser streaming for creating chat and real-time collaboration applications

- Programmatic control of streams (server scripting) for the creation of server-side playlists, synchronization of streams, smarter delivery adjusted to client connection speed, and application creation.

- Advanced monitoring and reporting on traffic and throughput.

Customers who do not want the hassle and expense of buying and maintaining server hardware and Flash Communication Server software can get all the benefits of streaming Flash video and MP3 files with the Flash Video Streaming Service. This service is a load-balanced, redundant deployment of Flash Communication Server over a reliable Content Delivery Network.
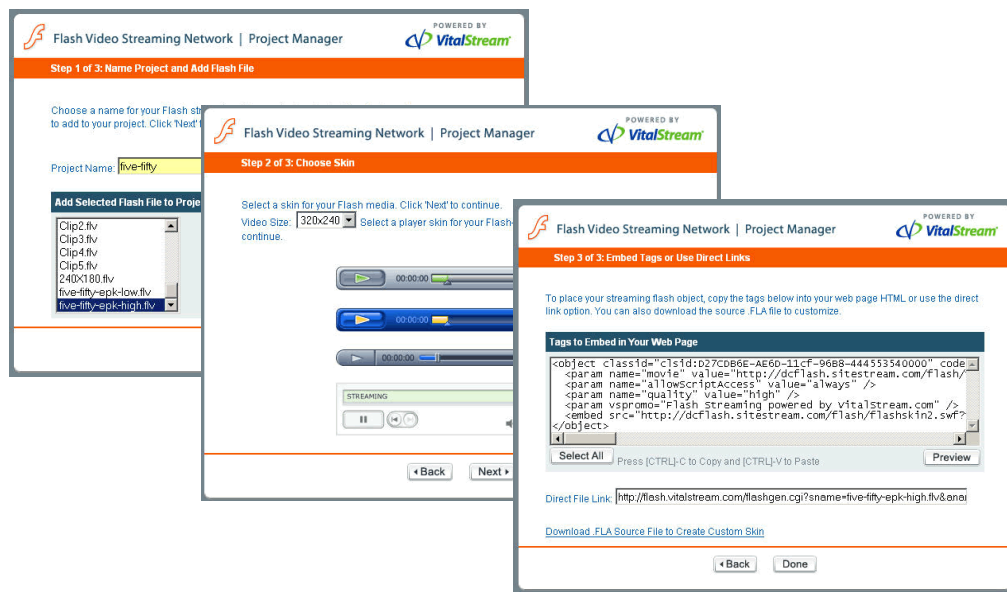
Figure 5: *A simple wizard walks you through publishing FLV files for streaming using the Flash Video Streaming Service.*

Table 2 provides a comparison of these three Flash Video delivery techniques.

Table 2: *Flash Video Delivery Techniques at a Glance*

	Embedded Video	**Progressive FLV**	**Streaming FLV**
Encoding	Video and audio is encoded on import into Flash using the Sorenson Spark codec. Alternately, FLV files (encoded elsewhere) can be imported and placed on the Flash Timeline (re-encoding is not necessary).	FLV files are encoded during export from various professional editing and encoding applications. Requires Flash Video Exporter, which is available only with Flash MX Professional 2004. FLV files can also be created by exporting the files from the Flash Timeline.	Same as Progressive FLV. In addition, bandwidth detection capabilities in streaming enable you to detect client connection and feed the appropriately encoded video. You can capture live video feeds from client-side webcams or DV cameras and control live encoding variables such as bitrate, fps, and video playback size

9

			programmatically.
File Size	SWF files contain both the video and audio streams as well as the Flash interface, resulting in a single, larger file size. SWF files can load each other, enabling you to break up individual video clips into multiple files.	SWF and FLV files are kept separated, resulting in a smaller SWF file size.	Same as Progressive FLV.
Timeline Access	When embedded in the Flash Timeline, video appears on individual keyframes and can be treated like any other object on the Stage. Ideal for creating interactions based on individual keyframes of video.	Video is played back only at runtime. Individual keyframes are not visible on the Flash Stage.	Same as Progressive FLV.
Publishing	Each time the Flash movie is published or tested the entire video file is re-published. Changes to video files require manually re-importing the files into the Flash Timeline.	FLV files are only referenced during runtime. Publishing does not require referencing the video file directly, and is much faster than the embedded video approach. FLV files can be updated or modified without changing the SWF files for a project.	Same as Progressive FLV. You can dynamically pull FLV from virtual locations, such as your SAN or the Flash Video Streaming Service CDN.
Frame Rate	Video frame rate and SWF movie frame rate must be the same.	The FLV video file can have a different frame rate than the SWF movie, allowing for separate timings between video and the Flash movie.	Same as Progressive FLV. Live video capture has programmable control over frame rate.
Action Script Access	Video playback and control is achieved by controlling the movie's playback on the Flash Timeline. Flash MX 2004 includes several behaviors for controlling video and audio playback.	The netStream command can be used to Load, Play, Pause, and Seek through multiple external FLV files.	Same as Progressive FLV. Server-side ActionScript can also be used to provide additional functionality such as synchronization of streams, server-side playlists, smart delivery adjusted to client connection speed, and more.
Components	No video-specific components.	Media Components (Flash MX Professional only) can be used to set up and display external FLV files together with transport controls (Play, Pause, and Search).	Media Components (Flash MX Professional only) can be used with video streamed from Flash Communication Server or Flash Video Streaming Service. Also, you can use Flash Communication Server Communication Components for streaming live and multiway video.
Web Delivery	The entire SWF file must be downloaded to the client and loaded into memory in order to play back video.	FLV files are progressively downloaded, cached, and then played from the local disk. The entire video clip need not fit in memory.	FLV files are streamed from Flash Communication Server, played on the client's machine, and then discarded from memory in a play-as-you-go method.

Performance	Audio and video synch is limited after approximately 120 seconds of video. Total file duration is limited to available RAM on the playback system.	Improved performance over embedded SWF video, with bigger and longer video and reliable audio synchronization. Provides best image quality, which is limited only by the amount of available hard drive space on the playback system.	Improved efficiency from a web delivery perspective, with optimal bit rate delivery on an as-needed basis to as many customers as necessary. Image quality limited to bit rates that can be delivered in real time.
Usage	Shorter video clips (less than 1 minute) that are smaller (less than 320 x 240), and have a lower frame rate (12 fps).	Longer video clips that are bigger (720 x 480 and greater) and have a higher frame rate (up to 30 fps).	Large quantities of video, very long video clips, and live and multiway streaming (such as webcam chat and live event broadcasts).
Compatibility	Flash Player 6 and 7	Flash Player 7	Flash Player 6 and 7

Using Live Video

The Flash video platform not only enables the use of recorded video—with Flash Communication Server, you can use live video as well. Flash Player 6 and higher include a video encoder that can capture video from a webcam or DV camera and publish that video to Flash Communication Server. The communication server can then publish that video to any Flash Player client (or any other Flash Communication Server) that subscribes to the stream.

A typical scenario for broadcasting a live video event would be as follows:

- The developer creates two applications (SWF files) with Flash – a *broadcaster* application and a *viewer* application.

- The video producer connects a DV or web camera to the broadcast machine through USB, Firewire, or a video capture card, and the broadcaster application (broadcaster swf) publishes the live video stream.

- Clients who want to view the live stream run the viewer application on their local machines.

This scenario is an extremely simple example of what you can do with Flash Communication Server. You can create even more compelling applications such as webcam chats, video messaging applications, and virtual classrooms. All of this is because Flash Communication Server provides multiway streaming to multiple clients; data streaming; video capture and recording on the server; and much more. Visit http://www.macromedia.com/software/flashcom/ for more information on using live video and advanced streaming features.

Creating Flash Video Files from within Third-Party Applications

Flash MX Professional 2004 includes Flash Video Exporter, which enables many third-party applications to encode video and audio directly into the

11

FLV file format. This ability to prepare FLV files from within third-party applications improves the workflow between video applications and your Flash project.

Flash Video Exporter achieves its seamless integration within third-party applications by leveraging the QuickTime export plug-in architecture. Flash Video Exporter is a plug-in to QuickTime that you can use with third-party applications that support exporting to the QuickTime file format. However, Flash Video Exporter does *not* produce QuickTime movies, but FLV files that contain Flash Player–compatible video and audio streams. QuickTime movies use the .mov extension; FLV files use the .flv extension.

After installing the Flash Video Exporter plug-in on your system, applications that support file export to the QuickTime file format will show Movie to Macromedia Flash Video (FLV) in their list of QuickTime export options. However, not all applications that support exporting to QuickTime support exporting to QuickTime Export Components. For a summary of leading third-party tools that support the Flash Video Exporter plug-in refer to the section "Supported Third-Party Applications" later in this paper. For full details on using each of these third-party tools, refer to the individual product manuals.

Using Flash Video Exporter in Demo Mode

For the Flash Video Exporter plug-in to work properly you must have a valid Flash MX Professional license installed and activated on your computer. If you are using Flash MX Professional in evaluation mode, or if you do not have a valid Flash MX Professional license installed, you can still use Flash Video Exporter in Demo Mode. While in Demo Mode you can use all features of the plug-in, but the FLV files created will show two one-pixel diagonal lines running across the resulting video as shown in Figure 5.
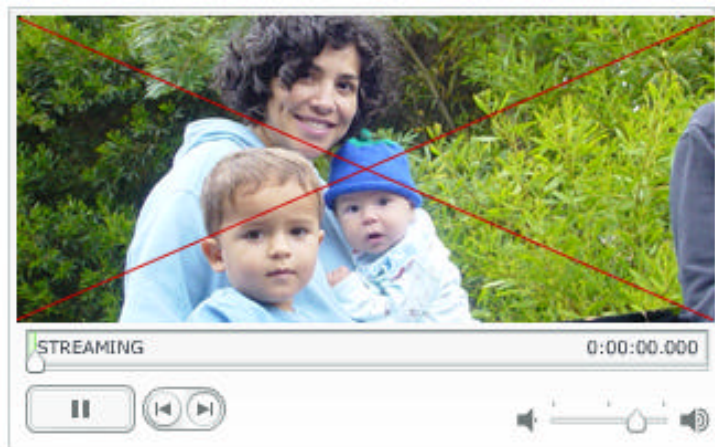

Figure 6: *Example of an FLV file encoded with Flash Video Exporter in Demo Mode.*

Supported Third-Party Applications

As of the writing of this document, the following applications support the Flash Video Exporter plug-in:

Editing and Effects Applications

- Adobe After Effects 5.5 and 6.0
- Apple Final Cut Express
- Apple Final Cut Pro (3.x and 4.x)
- Apple QuickTime Player 6.3
- Avid Media Composer
- Avid Xpress DV and Avid Xpress Pro

Batch Encoding Applications

- Anystream Agility
- Canopus Procoder
- Discreet Cleaner
- Discreet Cleaner XL

Using the Flash Video Exporter Plug-in

The Flash Video Exporter dialog box, shown in Figure 6, enables you to specify the video and audio quality of your FLV file during export from third-party editing and encoding applications. You can access this dialog box by choosing Options button from the host application's export dialog. (For more details see the section "Using Flash Video Exporter from within Third-Party Applications" later in this paper) The dialog box is divided into three groups: Video, Audio, and Other. Under Video you can specify the encoding options for your video; under Audio you can select the bit rate for your MP3 audio encoding; and under Other you can set your scaling and de-interlacing options. The following subsections explain in detail your encoding options.

Figure 7: *Choose Options in the host application's export dialog box to access the Flash Video Exporter dialog box.*

Encoding Method

The Encoding Method pop-up menu contains three items. Choosing the first two encoding methods produces a H263 format video stream, which is best suited for general-purpose video. The third encoding method consists of a separate video codec that uses a Macromedia proprietary codec optimized for screen recordings. You can choose from these three encoding methods:

- **Baseline (1 Pass)** uses a consistent bit rate to encode the video stream. The resulting file size is closely related to the value entered in the Limit Data Rate To text field, because the data rate will not vary on a frame-by-frame basis regardless of the complexity and changes in the video stream.

- **Better (1 Pass VBR)** allows the data rate to fluctuate based on the content in the video sequence, allocating more bits to areas of greater complexity. This setting produces better image quality, takes longer to process, and does not produce easily predictable file sizes (because the bit rate fluctuates depending on footage).

- **Screen Recording Codec** enables you to record screen operations to a lossless compression. When you select this option, all other video options become inactive (are grayed out). Only choose this encoding method when encoding footage that was created either in screen recording applications or by compositing graphic elements together in an editing or effects application. The codec records all pixel changes in the video stream, which is ideal if there is very little variation in the video (such as screen recordings of software application demonstrations), but will produce very large files when used with true video source footage.

Frames Per Second

The Frames Per Second text field enables you to specify the frame rate for the encoded FLV file. To maintain the temporal quality of the original source clip, use the same frame rate. For full motion NTSC use 29.97 fps; for PAL use 25 fps. If you lower the frame rate (which can significantly reduce the video data that needs to be encoded), the Flash Video Exporter plug-in will drop frames at a linear rate to achieve the new fps rate. Be sure to choose an evenly divisible fraction of the original frame rate, or you may introduce highly distracting "stepping" artifacts.

Table 3: *Divisible Frame Rates by Video Standard*

	FPS	Half	Third	Quarter
"Thirty"	30	15	10	7.5
NTSC	29.97	14.985	9.990	7.493
PAL	25	12.500	8.333	6.250
24p Progressive	24	12	8	6
Film	23.98	11.990	7.993	5.995

Note that 30 and 24 fps are not actually accurate frame rates in common use. (See the section "Technical Discussion of Video Fundamentals" for

additional information on the NTSC and PAL video standards.) To keep audio and video in synch when creating long FLV files (longer than a couple of minutes), you must use the accurate frame rates when downsampling the fps rate.

The pop-up menu to the right of the Frames Per Second text field enables you to cycle through the commonly used frame rates, but you can enter any frame rate you want in the Frames Per Second text field.

Quality

Use the Quality pop-up menu to specify the estimated data rate settings for encoding video at a low, medium, or high quality. This option controls the value in the Limit Data Rate To text field by computing an approximate data rate based on the file's frame rate (fps) and size (horizontal and vertical pixels). If you change the value in the Frames Per Second, Height, and Width text fields, Flash Video Exporter updates the value in the Limit Data Rate To text field to reflect the new quality settings. For example, if you change the frame rate from 24fps to 12fps, then the value in the Limit Data Rate To text field will be reduced by exactly 50 percent, because the required bit rate to maintain the given quality level is exactly half if you reduce the number of frames per second by half.

If the available Quality setting options do not produce the desired quality results with your particular source footage, select Custom, and then enter a higher data rate into the Limit Data Rate To text field.

Limit Data Rate To

This setting is the key "quality" variable and indicates the number of bits per second that are allocated to encode the video. Larger bit rates produce larger files with higher image quality.

The Flash Video Exporter dialog box displays data rate limit in kilobits or kilobytes per second. You can choose the unit you want from the pop-up menu to the right of the Limit Data Rate To text field. Some users might prefer to think in terms of kilobits while others think in terms of kilobytes. Those with a background in web development are familiar with bits and kilobits, because it is a web developer's duty to keep files as small as possible for quick downloads. Those with a background in print or video are accustomed to working with file sizes in the kilobyte, megabyte, and even gigabyte range and therefore might prefer to work with kilobytes. Simply put, there are 8 bits in a byte, 8 Kbits in 1 KB, 8 Mbits in 1 MB, and so on.).

When you enter a value x in the Limit Data Rate To text field, you are essentially telling Flesh Video Exporter, "do not make my final file size larger than x kilobits or kilobytes for every second. If you have a 10 second clip and enter a value of 100 KB/sec, you'd end up with a clip no larger than 1,000 KB, or 1 MB. In order to keep down the file size encoders will sacrifice image quality. To keep the image quality consistent, you can scale down the clip so fewer pixels per second are represented, or lower the frame rate so less frames per second are displayed.

When exporting files, you must find the right balance between image size, file size, and frame rate. For best results, consider your target delivery

medium. For Internet delivery most FLV files are downloaded progressively to the client computer and then played back from the local hard disk. This technique provides a wide gamut of bit rates for delivery. For streaming FLV files, client connection speeds limit the bit rate that you can use.

Table 4 shows the approximate total download times for a one- and five-minute video clip encoded at 500 kilobits and 2,000 kilobits. Both clips assume no audio (MOS).

Table 4: *Downlaod Times for Video Clips*

	Kbits	500 Kbit Video		2,000 Kbit Video	
		1 min	5 min	1 min	5 min
T3	1500	20 sec	2 min	80 sec	7 min
T1	768	39 sec	3 min	3 min	13 min
DSL	512	59 sec	5 min	4 min	20 min
DSL	384	1 min	7 min	5 min	26 min
Dialup	56	9 min	45 min	36 min	179 min

Table 5 lists the approximate data rates that you can use with Flash Communication Server to reliably stream video to target users on various connection speeds. These data rates include a 10-percent headroom to allow for network traffic.

Table 5: *Data Rates for Streaming Video*

Connection	Connection Rate (Kbps)	Recommended Maximum Bit Rate
DSL	1024	800 kbps
DSL	512	450 kbps
DSL	384	350 kbps
DSL	256	225 kbps
Dialup	56.6	40 kbps

Keyframes

The Keyframes pop-up menu enables you to choose between automatic or custom keyframing. The Keyframe value indicates to the encoder how often to re-evaluate the video image and record a full frame, or keyframe, into the FLV file. Between keyframes the encoder approximates the full frame by estimating the full value of all pixels on the screen, principally by comparing multiple frames and eliminating information that is redundant.

The Keyframe value can be any number up to 100. When you select Auto, Flash Video Exporter chooses a keyframe rate that is a multiple of the

current frame rate. The Auto setting is a good starting point for all video footage. If your footage has a lot of scene changes or rapidly moving motion or animation, then the overall image quality may benefit from a lower keyframe value. In general, a higher keyframe rate produces better image quality because bits are not wasted describing the areas of an image that remain unchanged from frame to frame.

There is a very important dependency on the keyframe rate, which refers to the ability of Flash Player to seek (fast forward, rewind) through an FLV file. Flash Player can only frame advance to keyframes, so if you want to be able to skip around to different places and pause the frames, you must use a lower keyframe value. If you want to be able to advance to every frame in the FLV file, use a keyframe value of 1. When you reduce the keyframe value, you must raise the data rate for the Flash video in order to maintain comparable image quality.

Motion Estimation

The Flash Video Exporter encoder analyzes the differences and similarities between adjacent frames, and allocates bits depending on this information. Using the Motion Estimation pop-up menu, you can choose between Faster, which results in faster estimation of motion at the sacrifice of quality, or Better, which results in a more accurate estimation of motion at the sacrifice of speed.

Audio Bitrate

The Bitrate pop-up menu under Audio enables you to set the bit rate of your MP3 audio stream. Full audio tracks, such as music and significant background noise, require a higher bit rate. Simple dialogue, such as with a "talking head" shot, can be compressed to a much higher degree. Higher bit rate settings are encoded in stereo, while lower bit rate settings are encoded mono.

Resize To

The Resize To pop-up menu enables you to scale your footage on output. The pop-up menu lists a number of common preset sizes, but you can enter a size of your choice in the Width and Height text boxes. You can also designate the output size by entering a percentage value of the source size rather than an arbitrary pixel value.

To keep the aspect ratio (the ratio of width to height) of the source video file, select the Lock Aspect Ratio check box. Leave this box unchecked, if you want to export your video at a different aspect ratio than the source file.

In some third-party applications the source file's image size may not automatically show up in the Flash Video Exporter dialog box. In these cases, if you want to constrain the proportions you enter the source file's pixel dimensions, select the Lock Aspect Ratio check box, and then resize the image.

If your source video is made up of non -square pixels, such as D1 or DV video, use the scale function to scale the image non -proportionately down to

a 4:3 aspect ratio. For example, if your source file is 720 x 480 (DV NTSC), then scaling to 320 x 240 (correct aspect ratio) will require a non - proportional scale (greater change in X axis than Y).

De-interlacing

When encoding interlaced source footage (any NTSC or PAL footage shot with a video camera) you will most likely want to remove one of the fields, or *de-interlace* the footage, prior to encoding. Many third-party applications offer this capability within their own interface, but you can also choose this option in the Flash Video Exporter dialog box. Using the De-interlacing pop-up menu, you can choose to eliminate the upper or lower field. For more information, see the section "Interlaced and Progressive Video" earlier in this paper.

Using Flash Video Exporter from within Third-Party Applications

This section provides brief descriptions of how you can access the Flash Video Exporter dialog box from within third-party applications.

Adobe After Effects

As of version 6.0 of Adobe After Effects Flash Video Exporter is not available in After Effect's standard rendering cue. Instead, it is listed in the File | Export submenu. To access the Flash Video Exporter dialog box, select the Export command from the File menu (see Figure 7).
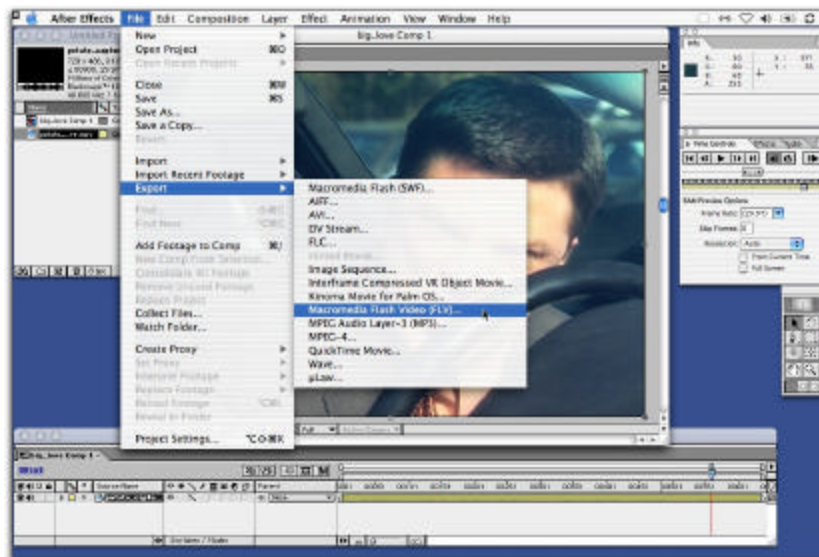

Figure 8: *Accessing the Flash Video Exporter dialog box in Adobe After Effects 6.0.*

Important Note: When exporting files from After Effects, the current composite's resolution dictates the output of the FLV file. For example, if you have a D1 720 x 486 comp set to Quarter resolution and export to the

FLV file format, Flash Video Exporter reads the file as 180 x 121. If you then choose 720 x 486 in the FLV output size, you will get a scaled-up, pixilated movie. So make sure to set your After Effects comp to Full resolution before exporting (see Figure 8).
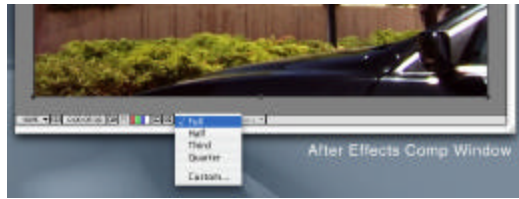

Figure 9: *Be sure your After Effects comp is set to Full resolution before exporting your video.*

Apple Final Cut Pro

With Final Cut Pro 4.0 you can export to Flash Video by choosing Export | QuickTime Conversion from the File menu. When you are working with earlier versions of Final Cut Pro and Final Cut Express choose File | Export | QuickTime to open the export dialog box. To finish the conversion, choose Macromedia Flash Video (FLV) from the Format pop-up menu (see Figure 9), and then click the Options button to proceed to the Flash Video Exporter dialog box.
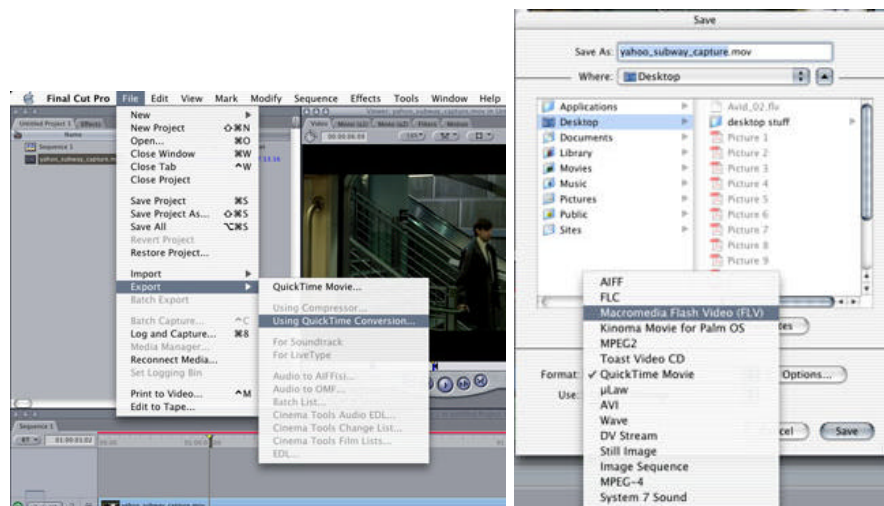

Figure 10: *Accessing the Flash Video Exporter dialog box in Final Cut Pro.*

Note: Final Cut Pro shows the current window's pixels size as the size of the video clip, so if you want to keep this size you must manually enter the Height and Width settings in the Flash Video Exporter dialog box. If you want to scale down proportionately, you must first enter the source clip's file size, then select the Lock Aspect Ratio check box to constrain proportions, and then resize the video.

Apple QuickTime Player (Mac OS and Windows)

The Flash Video Exporter plug-in is based on Apple's QuickTime architecture. QuickTime Player enables you to open video files saved in

19

various video formats. When you upgrade to QuickTime Pro you can also export video files to a different format.

To export a movie to the FLV file format, open the file in QuickTime Player and choose Export from the File menu. This opens the Save Exported File As dialog box. Select Movie to Macromedia Flash Video (FLV) from the Export pop-up menu and then click the Options button to open the Flash Video Exporter dialog box (see Figure 10).
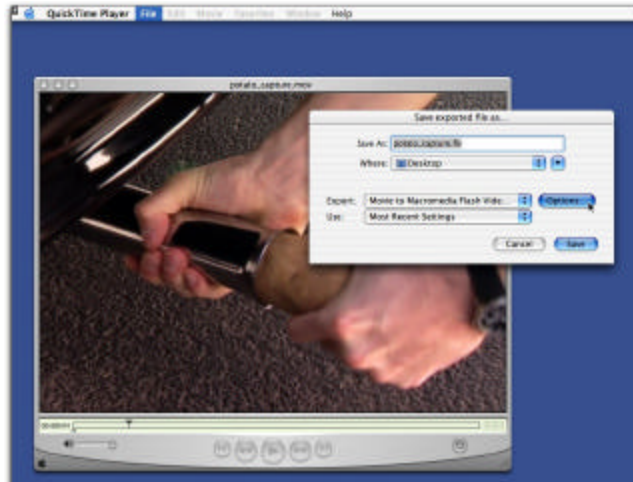

Figure 11: *Accessing the Flash Video Exporter dialog box in Apple QuickTime Player.*

Avid Media Composer and Xpress

Accessing the Flash Video Exporter dialog box works the same in both Media Composer and Xpress DV / Xpress Pro applications: Select Export from the File menu and then click the Options button in the export dialog box to open the Export Settings dialog box. From the Export As pop-up menu, select Macromedia Flash Video (FLV) (see Figure 11), and then click Format Options to launch the Flash Video Exporter dialog box.
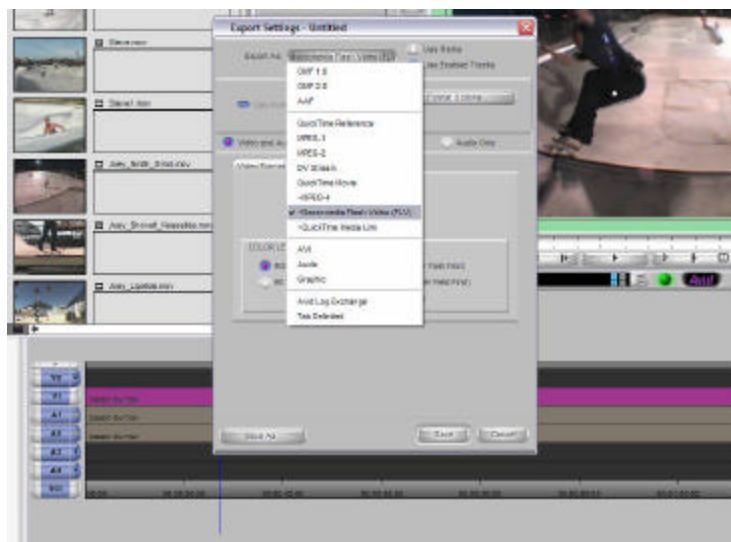

Figure 12: *Accessing the Flash Video Exporter dialog box in Avid Media Composer.*

Anystream Agility

To export to the FLV file format from within Anystream Agility, create a new job profile in the Main window or open an existing one. Select Add QuickTime Encode from the Encoder menu or press Ctrl+T. Under QuickTime Format Type, select Other Formats from the QuickTime Format drop-down list. Next, select Macromedia Flash Video (FLV) from the Format drop-down list under Target Settings (see Figure 12).
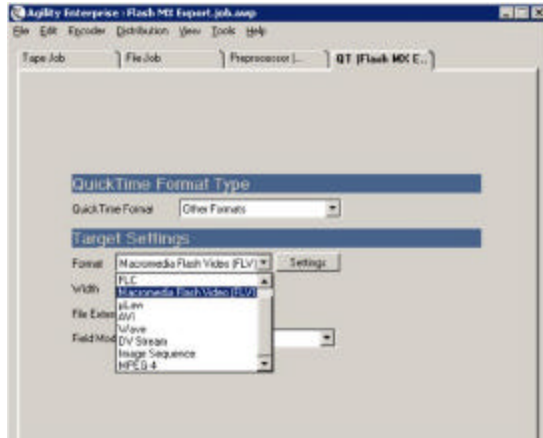

Figure 13: *Selecting the Macromedia Flash Video (FLV) file format in Anystream Agility.*

Under Target Settings, you can also specify the width, height, and field mode of your video (see Figure 13).

Note: If you are specifying width and height settings under Target Settings remember *not to* reapply a scaling factor in the Flash Video Exporter dialog box.
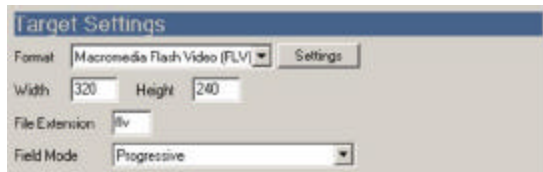

Figure 14: *Under Target Settings you can specify the width, height, file extension, and field mode parameters of your video.*

Canopus Procoder 2.O

To export to the FLV format from within Canopus Procoder select the Target Tab, click the Add button, and then select QuickTime Exporter (see Figure 14). From the Target Parameters tab, under Destination, select Macromedia Flash Video (FLV) from the drop-down list. To access the Flash Video Exporter dialog box, click the Modify Exporter Settings button.
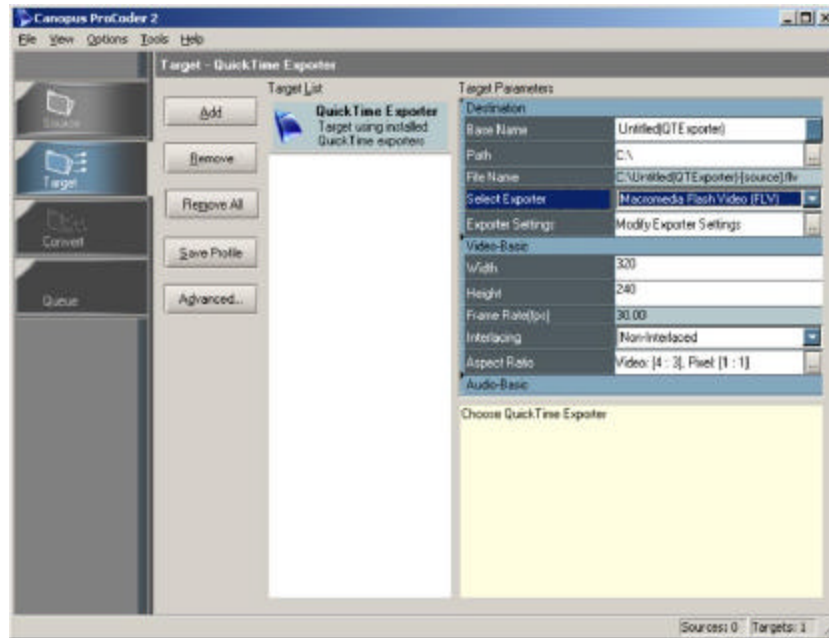
Figure 15: *Accessing the Flash Video Exporter dialog box in Canopus Procoder.*

Discreet Cleaner (Mac OS)

Discreet Cleaner enables Flash MX Professional 2004 users to encode multiple FLV files on unattended computers. Like other file formats such as QuickTime or JPEG, the Macromedia Flash Video (FLV) file format shows up in the Format list on the Output Tab of the Cleaner Settings dialog box (see Figure 15).
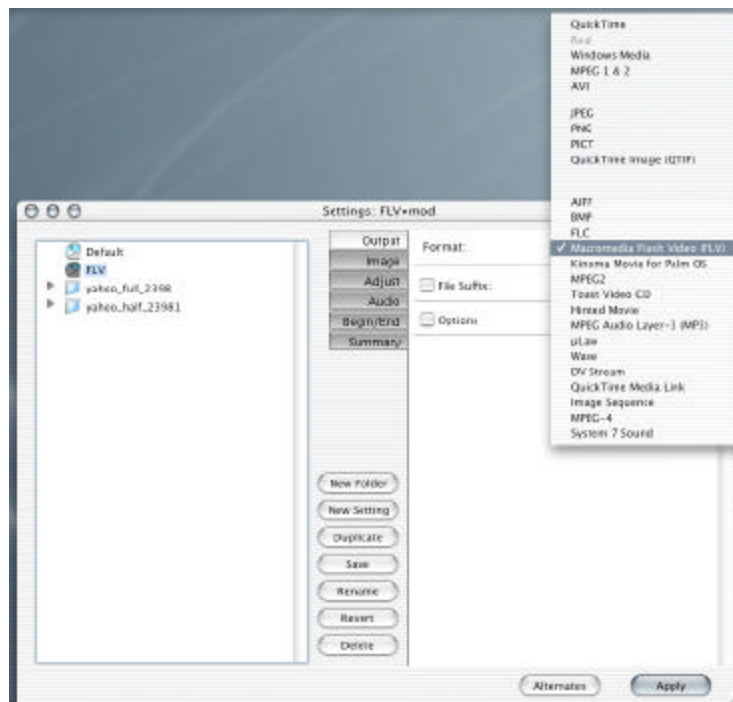
Figure 16: *Accessing the Flash Video Exporter dialog box in Discreet Cleaner.*

To access the Flash Video Exporter dialog box, choose Macromedia Flash Video (FLV) from the Format pop-up list, select the Options check box, and then click the Set button (see Figure 16).
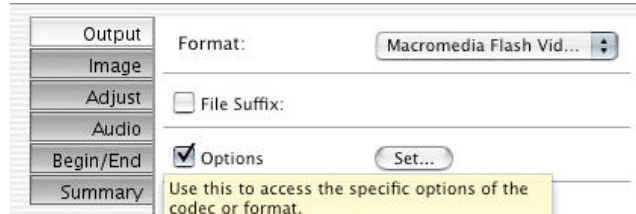

Figure 17: *Accessing the Flash Video Exporter dialog box in Discreet Cleaner.*

Discreet Cleaner has many options to preprocess your image, including color correction, resizing, de-interlacing, cropping, pulldown removal, and so on. These settings are applied before the file is converted to the FLV file format.

Note: When exporting to the FLV file format, be careful not to apply duplicate scaling or de-interlacing passes by using both the Cleaner and Flash Video Exporter settings—this creates unwanted results such as applying to scale transformations, resulting in a smaller size than desired.

Discreet Cleaner XL (Windows)

Using Discreet Cleaner XL and Flash MX Professional 2004, you can encode multiple FLV files on unattended computers. To get started, select an existing profile, then click Encoder Setting to open the Encoder Setting Editor dialog box where you can select Macromedia Flash Video (FLV) under Encoding Format (see Figure 17).
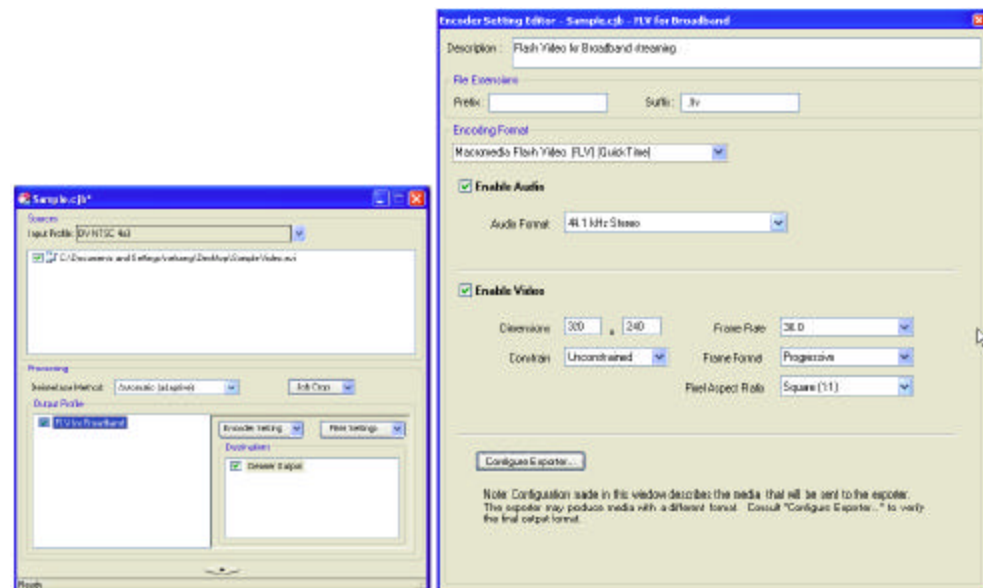
Figure 18: *Accessing the Encoder Setting Editor dialog box in Discreet Cleaner XL.*

Note: When exporting to the FLV file format, be careful not to apply duplicate scaling or de-interlacing passes by using both the Cleaner and Flash Video Exporter settings—this will create unwanted results such as applying to scale transformations, resulting in a smaller size than desired.

Other Third-Party Products That Support the FLV File Format

The following third-party products have built-in support for the Flash Video (FLV) file format:

- Sorenson Squeeze
- WildForm Flix
- Pinnacle Systems Edition 5

For more information on how to export to the FLV file format, refer to their user manuals.

Macromedia Flash MX 2004 Video FAQ

What standard is Flash Video compression based on?

Flash Video is based on the H.263 video compression standard. It's based on the version of H.263 dated May 1996, sometimes referred to as H.263v1. This is distinct from the revised version of H.263, dated February 1998, sometimes referred to as H.263v2 or H.263+.

The H.263 video format supported by Flash differs slightly from H.263. For the most part, it is a subset of H.263, with some features removed and others added a shown in the following lists.

The following H.263 features have been removed from the Flash Video format:

- GOB (group of blocks) layer
- Split-screen indicator
- Document camera indicator
- Picture freeze release
- Syntax-based arithmetic coding
- PB-Frames
- Continuous presence multipoint
- Overlapped block motion compensation

The following non-H.263 features have been added to the Flash Video format:

- Disposable frames (difference frames with no future dependencies)
- Arbitrary picture width and height up to 65,535 pixels
- Unrestricted motion vector support is always on
- A deblocking flag is available to suggest the use of a deblocking filter

Where can I get more information about the Flash Video file format?

Detailed technical information about the FLV file format is published as part of the SWF specification at www.macromedia.com/software/flash/open/licensing/fileformat/. An open source implementation of the FLV video format is available at ffmpeg.sourceforge.net/.

What are the best pixel dimensions for transporting or encoding data based on the H.263 standard? Would there be any benefits to encoding the video at: 128 x 96, 176 x 144, 352 x 288, 704 x 576, 1408 x 1152?

Flash Video supports an extension to H.263 that supports frames with any dimensions. However, there is probably a small benefit to frame sizes that are a multiple of 8 x 8 pixels.

What improvements to video performance have been made with Macromedia Flash Player version 7?

Flash Player 7 increases playback performance for video significantly by allowing full speed playback of much larger and longer video files. Flash Player 7 also adds support for progressive download video, allowing FLV files to be loaded and played back at runtime.

How does Flash Video work with QuickTime?

The Flash Video Exporter plug-in (included with Flash MX Professional) enables applications that support QuickTime movies to encode/transcode directly video and audio into the Flash Video (FLV) file format. Additionally, Flash MX and Flash MX Professional 2004 support importing QuickTime movies, which are then encoded into FLV and placed in the Flash Timeline. Flash does *not* support playing back QuickTime movies, such as those encoded in MPEG2 or MPEG4.

Can Flash Player play back Windows Medi a, QuickTime, or Real video files?

No. Flash Player does not support playback of video files created with other video codecs such as those used in Windows Media Player, Real Player, or QuickTime Player. These files can be transcoded into the Flash Video file format (FLV) with third party products such as Sorenson Squeeze, Agility from Anystream, or Cleaner from Discreet.

Can I stream Flash Video?

Yes, with Macromedia Flash Communication Server (available separately). See separate FAQ section on Flash Commun ication Server.

Can I use Flash Video on devices?

Yes, devices that have Flash Player 6 or later support video playback and video streamed from Flash Communication Server. Earlier versions of Flash Player, including Flash Lite, do not support video playback features.

Can I use Flash Video on CD/DVD ROM?

Yes, Flash Video files can be played back from fixed media such as CD-ROM or DVD-ROM. For greater control and enhanced capabilities when using fixed media, consider Macromedia Director MX.

Can I encode Flash Video using batch-encoding tools?

Yes, batch encoding tools such as Discreet Cleaner and Anystream Agility support the Flash MX Professional Flash Video Exporter plug-in and therefore can encode Flash Video files. Batch encoding tools can be used to encode multiple video clips during one batch process.

Can I encode Flash Video using nonlinear editing tools?

Yes, digital video editing tools such as Avid Media Composer and Xpress, Apple Final Cut Pro, and Pinnacle Edition support the Flash MX Professional Flash Video Exporter plug-in and therefore can encode Flash Video files.

What is the difference between Flash MX and Flash MX Professional 2004 in terms of video features and capabilities?

Flash MX Professional 2004 adds significant video capabilities: Media Components and Flash Video Exporter accelerate development and simplify production workflow; slides and special media behaviors facilitate creating advanced video presentations.

Flash Communication Server Streaming Questions

How does Flash Communication Server integrate with Flash MX 2004?

Flash Communication Server MX is tightly integrated with Flash MX 2004 to enable developers to create and deploy compelling, interactive audio/video applications.

Flash Communication Server MX provides a complete set of authoring tools that install in the Flash MX 2004 environment, including a number of ActionScript statements, a new Communications App Inspector panel, a NetConnection Debugger panel, and a number of prebuilt communications components.

Using these authoring tools, developers use Flash MX 2004 to author their applications (SWF files). When a Flash Communication Server application is viewed in Flash Player 6, a persistent connection is made back to Flash Communication Server. Audio, video, and data can then be streamed over the persistent connection.

What are some of the benefits to using Flash Communication Server MX for audio/video delivery?

Flash Communication Server MX provides a number of advantages for audio and video delivery, including:

- **Multiuser, multiway streaming:** Built-in multipoint control enables developers to take advantage of multiway, multiuser streaming to create a whole new class of audio and video applications such as video on demand, live event broadcast, webcam chat, and real-time collaboration applications.

- **Video capture:** Developers can easily create applications that capture video from standard webcams and DV cameras.

- **Faster start and better buffering:** Buffering enables videos to be seen immediately rather than having the end-user wait.

- **Programmable stream control:** A server scripting environment enables development of audio and video applications with unprecedented levels of functionality and interactivity.

- **Data Streaming:** The distributed data synchronization capabilities in FCS enable developers to create applications that use real-time data such as close captioning, stock tickers, sports scores, or multiuser games.

What file formats does Flash Communication Server MX stream?

Flash Communication Server MX can stream both FLV and MP3 file formats.

What standard is the Audio compression based on?

For playback, Flash Player supports MP3, ADPCM, and a proprietary voice codec. For capture, Flash Communication Server MX supports the proprietary voice codec. The voice codec can be licensed from a company called Nellymoser.

How can I send the best video depending on the client connection?

Because Flash Communication Server-based applications establish a persistent connection between the client and server, developers can determine the client's connection speed and deliver the most appropriate video. For example, for recorded video, the developer can choose to deliver the video with the bit rate closest to the client's connection rate. For live video, the developer can choose to alter video stream properties such as video size, frames per second, quality, or bandwidth settings to deliver appropriate video.

How can I send Flash Video most effectively over my network?

Because only bits that are viewed are sent to the client, streaming in itself offers a very effective form of video delivery in terms of network bandwidth. In addition, Flash Communication Server MX also provides for server-to-server stream passing (or splitting) to enable a branch and leaf style cluster to be configured. This way, the same live or recorded video stream can be served up from multiple servers running Flash Communication Server MX on the network so no one machine becomes overburdened. This enables developers to create scalable server configurations to handle large numbers of clients and supports failover and redundancy servers. For more information about how to set up Flash Communication Server MX to support a multiple server environment go to www.macromedia.com/desdev/mx/flashcom/articles/scalable.html

What kind of capacity backbone is needed for streaming video over the Internet with the Flash Communication Server?

The capacity that you need to stream your video broadcast depends on the number of people that will be simultaneously viewing your video and the size of your video stream.

You can get a simple estimate of the bandwidth you will need to stream your video by multiplying the number of simultaneous viewers that will be watching your video times the bandwidth at which they will be viewing it. For example, if your video is being streamed to 4 viewers and each video broadcast is at 200 Kilobits per second (kbps) then the backbone you'll need is 200 kbps x 4 = 800 kbps.

For more information download the white paper on this topic at www.macromedia.com/software/flashcom/productinfo/editions/fcs_whitepaper_bandwidth.pdf.

Where can I find information about the pricing and different editions of Flash Communication Server?

You can find pricing information and edition details about Flash Communication Server at www.macromedia.com/software/flashcom.

Macromedia Director MX and Flash Video

What additional video capabilities does Macromedia Director MX add to a Flash MX project?

Director MX adds the following video capabilities to a Flash MX project:

- Full-screen, high performance video in a variety of formats (for example, QuickTime, RealVideo, FLV)
- Additional Media types (for example, BMP, GIF, and more)
- Extensibility (device control, custom input devices, and so on)
- Interactive real-time 3D
- Higher performance animation for games and simulations
- Fine-grain memory management
- Tools for deploying to CD-ROM, DVD-ROM, or kiosks

Where can I find additional information about Macromedia Director MX and Flash MX Professional?

Check out these resources:

- On using Flash MX and Director MX together:
  http://director.breezecentral.com/p82592013/
- On using digital video in Director:
  www.macromedia.com/software/director/special/tours/video/
- Other Director tutorials:
  www.macromedia.com/software/director/special/tours/